Fascículo 7





OpenOffice.org Base

La ofimática para todos

Cuadernos de formación tecnológica



7. OpenOffice.org como gestor de bases de datos

Una base de datos es un conjunto de datos que están organizados entre sí y que pueden consultarse.

El programa que permite el almacenamiento de los datos y que facilita su consulta se llama sistema gestor de bases de datos.

Ejemplos típicos de bases de datos pueden ser el catálogo de libros de una biblioteca, las películas presentes en un videoclub o las calificaciones de los alumnos en un Instituto.

En la actualidad existen diversos modelos de bases de datos como, por ejemplo, el jerárquico, el de red y el relacional, aunque éste último es el que ha adquirido mayor popularidad y es el que más se utiliza. La idea fundamental de este modelo de base de datos recae en el uso de relaciones entre tablas que contienen datos.

Una tabla es el sistema que permite almacenar datos de una determinada entidad. Cada tabla puede contener varios atributos relacionados con la entidad que describen. Llamamos registro a cada fila de información que contiene la tabla.

Imaginemos que queremos crear una base de datos para la gestión de las películas presentes en un videoclub, se podrían crear las tablas siguientes:

– Tabla **Películas**: donde se almacenan los datos de la película y, por tanto, podríamos definir los atributos de *Título*, *Año*, *Duración*, *Director*, *Temática*, Actor *principal*, entre otros.

– Tabla **Actores**: donde se almacena información de los actores como el *Nombre*, *Apellidos*, *Año de nacimiento*, entre otros.

– Tabla **Directores**: donde se almacena información sobre los directores de las películas como el *Nombre*, *Apellidos*, *Año de nacimiento*, *Temática*, entre otros.

Para cada atributo de la tabla se debe configurar el tipo de datos que contendrá. Por ejemplo, para el atributo *Año* de la tabla **Películas**, lo más adecuado sería utilizar como tipo de campo *Fecha*. En cambio, para el campo *Duración*, lo correcto sería uno de tipo numérico y para el campo *Nombre*, uno de tipo cadena de caracteres.

Cada tabla puede contener una llave primaria que identifica, de forma inequívoca, cada registro de una tabla. Por tanto, el valor de una llave primaria no se puede repetir entre las filas de una tabla.

Podríamos utilizar un campo DNI/CIF como llave primaria en una tabla destinada a almacenar información de clientes o personas, ya que todos tenemos uno distinto. Estas llaves que tienen significado, se denominan llaves naturales.

Es importante destacar que, en este modelo, el orden en que se almacenan los datos carece de importancia (a diferencia de otros modelos de bases de datos), ya que esta información se presentará al usuario a través de consultas que permiten ordenar los datos de la manera más conveniente.

El lenguaje más habitual para construir las consultas en las bases de datos relacionales es el SQL (*Structured Query Language* o Lenguaje Estructurado de Consultas), un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales, aunque para los usuarios más inexpertos existen otros métodos que no requieren el aprendizaje de este lenguaje.

Una base de datos acostumbra a tener diversas tablas relacionadas entre sí. En el caso anterior, podríamos relacionar el atributo *Actor principal* de la tabla **Películas** con la llave primaria de la tabla **Actores**. De esta manera es como podemos relacionar diversas tablas.

Las relaciones más usuales entre tablas son las siguientes:

- **Relación de 1 a 1**: para cada registro de una tabla corresponde sólo un registro en otra tabla y viceversa.

- **Relación de 1 a muchos**: éste es el tipo de relación más habitual entre tablas. Cada registro de una tabla se puede relacionar con muchos otros registros de otra tabla.

  En el caso expuesto anteriormente existe una relación de *1 a muchos* entre la tabla **Directores** y **Películas**, y es que cada *Director* puede haber dirigido muchas *Películas* y, en cambio, una *Película* normalmente ha sido dirigida por un solo *Director*.

- **Relación de muchos a muchos**: esta relación existe en situaciones en las que cada registro de una tabla se puede relacionar con muchos otros registros de otra tabla y viceversa.

7.1 OpenOffice.org Base

El módulo de OpenOffice.org (OOo) destinado a la gestión de bases de datos es Base. Este programa es una de las grandes novedades que incorpora la versión 2.0 de OOo, ya que no existía como tal en la versión anterior, y puede obtenerse en español desde la web del proyecto *http://es.openoffice.org*.

OpenOffice.org Base es comparable a otros sistemas gestores de bases de datos para escritorios personales como Microsoft Access de Microsoft Office.

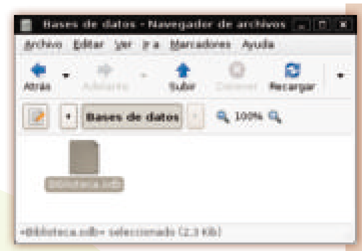
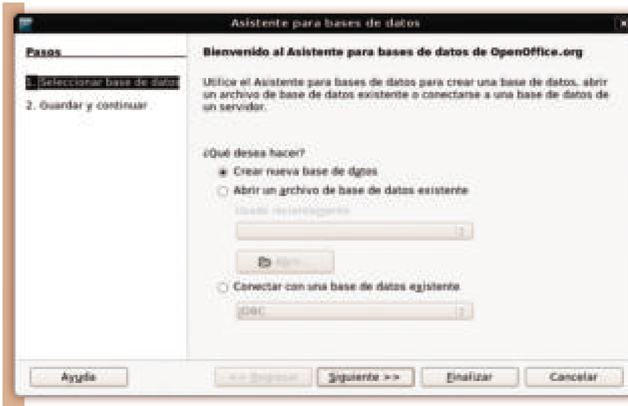


Figura 1. Fichero general con OpenOffice.org Base.

7.2 Creación de una base de datos

Crear una nueva base de datos desde cero utilizando OpenOffice.org es bastante sencillo. Simplemente, en el momento de iniciar la aplicación, debemos seleccionar la opción **Crear nueva base de datos** (**Figura 2**). El tipo de base de datos por defecto será un fichero nativo de OpenOffice.org en formato *.odb* que utiliza el motor HSQL.



A continuación, el asistente nos pregunta si deseamos registrar la base de datos en OpenOffice.org. Si escogemos **Sí, registrar la base de datos** podemos acceder a esta base de datos desde otros módulos de OpenOffice.org como Writer o Calc.

Figura 2.
Asistente para la creación
de una nueva base de datos.

Conexión a una base de datos remota

OpenOffice.org Base permite también la conexión a otros gestores de bases de datos diferentes a través de diversos mecanismos.

Permite, por ejemplo, acceder a una base de datos MySQL a través de los conectores ODBC o JDBC, e incluso a nuestra libreta de direcciones de contactos presentes en nuestro programa gestor de correo electrónico. Para hacerlo, seleccionamos la opción **Conectar con una base de datos existente** en el asistente que aparece al inicio y seguimos los pasos que se indiquen, que son diferentes dependiendo del tipo de base de datos escogida.

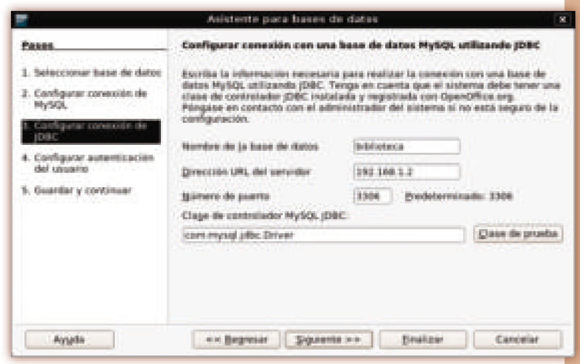


Figura 3. Ejemplo de conexión a una base de datos MySQL.

Trabajo con tablas

Una vez hemos escogido la base de datos en la que queremos trabajar, ya sea una base de datos incrustada en un fichero de OpenOffice.org o en una base de datos remota (como MySQL u Oracle), el programa nos permite la creación de una nueva tabla destinada a almacenar datos.

Antes de crear la nueva tabla, es importante tener claro cuál es la información que debe almacenar esta tabla y, por tanto, los atributos que describen esta información. También, definir qué atributo actuará como **llave primaria** de la tabla.

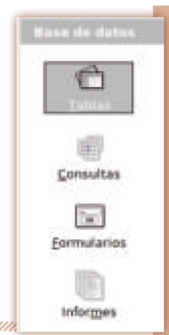
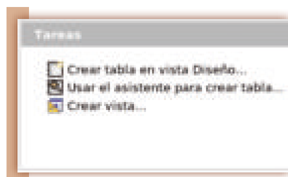


Figura 4. Trabajo con **tablas**.



El programa nos ofrece dos posibilidades para la creación de tablas, bien utilizar el **asistente** o directamente a través de la **vista Diseño**. En los apartados siguientes veremos cada uno de los métodos.

Figura 5. **Tareas** disponibles en el apartado de trabajo **Tablas**.

Utilizando el asistente

Si hacemos clic en la opción **Usar el asistente para crear una tabla...** aparecerá un asistente que nos permitirá crear en sólo 4 pasos una nueva tabla. Éste es el método más sencillo y, por tanto, el más indicado para usuarios noveles.

Paso 1: Selección de campos.
El primer paso consiste en elegir los atributos que contendrá nuestra nueva tabla. El asistente ofrece ya algunas tablas y campos típicos clasificados en dos categorías diferentes (negocios y personal).

Paso 2: Tipo y formato para los campos.
En el segundo paso seleccionamos el tipo y el formato para cada atributo escogido en el paso anterior (ver **Figura 6**). En este cuadro se nos pide introducir:

1. El **Nombre del campo** ❶.
2. El **tipo de campo** ❷ que indica si este campo contendrá textos, números, datos, etc.
3. Si se **necesita una entrada** ❸, refiriéndose a si será obligatorio introducir datos en este campo.
4. Y finalmente, determinar la **longitud** ❹ máxima para el campo. Por ejemplo, para almacenar un DNI sería suficiente con 9 caracteres, y así se puede ahorrar espacio en el disco.
5. Con los botones ❺ y ❻ podemos suprimir o añadir un nuevo campo, si fuera necesario.

Dependiendo del tipo de datos que hayamos escogido para el campo en cuestión, nos aparecerán algunas opciones diferentes. Por ejemplo, si se trata de un **Entero**, aparecerá la opción **Valor automático** a la que podremos responder **Sí** o **No**. Si escogemos la primera opción, el mismo programa introducirá automáticamente un valor en este campo si nosotros no lo hacemos.

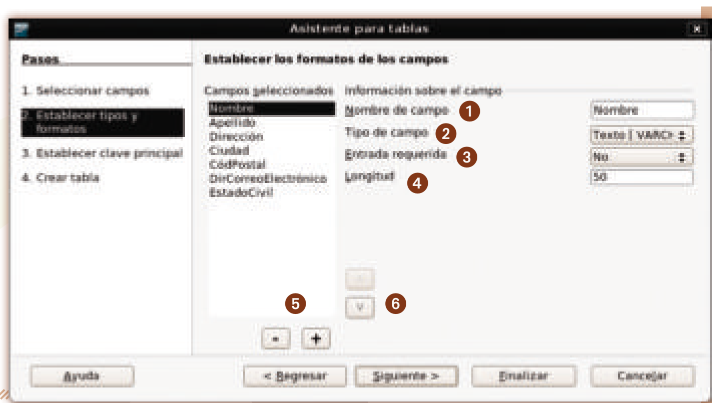


Figura 6.
Segundo paso en el asistente para la creación de tablas.

En cambio, si se trata de un campo para **Decimales**, nos aparecerá la opción **Cifras decimales** donde podemos indicar al programa la cantidad de cifras decimales que nos serán necesarias.

Paso 3: Definición de la llave primaria.
El tercer paso nos servirá para determinar cuál será el campo que actuará como llave primaria de la tabla. El asistente nos permite tres opciones:

- Que el mismo programa añada un nuevo atributo a la tabla que actúe como llave primaria.
- Que se utilice un campo ya existente como llave primaria.
- O definir una llave primaria como una combinación entre diversos campos.

Recordemos que el valor de una llave primaria no se puede repetir entre las filas de una misma tabla, ya que debe ser siempre diferente.

Paso 4: Creación de la tabla.
Finalmente, procedemos a la creación de la tabla. Sólo nos queda especificar un nombre y decidir la acción a ejecutar a continuación.

Utilizando la vista de diseño

El otro método que nos ofrece el programa para la creación de tablas es utilizar la vista de diseño. Éste es el método preferido para usuarios más expertos, puesto que es más ágil y potente que el que acabamos de ver.



Figura 7. Ejemplo de creación de tablas utilizando la vista de diseño.

La herramienta presenta una rejilla donde cada fila representa una columna de la tabla resultante. Observamos una columna que indica el **Nombre del campo**, otra que indica el **Tipo de campo** y, finalmente, una **Descripción** para el campo. En **Tipo de campo** aparece una lista desplegable para escoger qué tipo de datos almacenará este campo. A continuación, algunos de los tipos de datos más utilizados son estos:

- **Text (VARCHAR)**: para almacenar cadenas de texto.
- **Número (NUMERIC)**: para almacenar números.
- **Data (DATE)**: para registrar fechas del calendario.
- **Hora (TIME)**: para almacenar horas del reloj.
- **Sí/NO (BOOLEAN)**: para registrar un *Sí* o un *No*.
- **Memo (LONGVARCHAR)**: para almacenar textos largos.

En la parte inferior de la pantalla se especifican las **Propiedades del campo**.

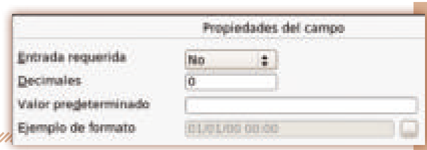
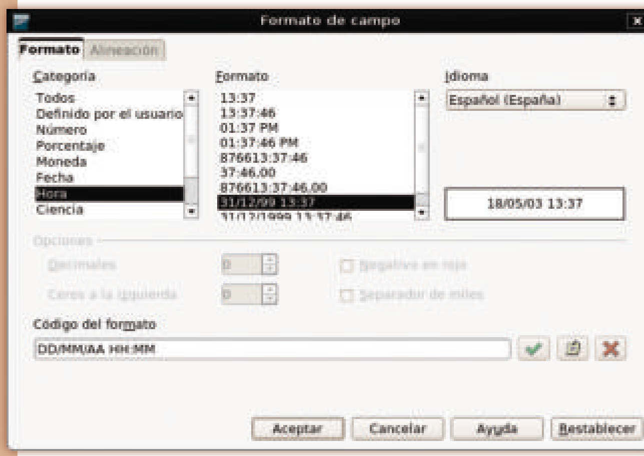


Figura 8.
Propiedades del campo VARCHAR para textos.



Estas propiedades son casi las mismas que las opciones descritas en el paso 3 del apartado **Utilizando el asistente**. Pero se incluye una opción **Ejemplo de formato**, donde es posible determinar qué formato seguirán los datos que se ubiquen en este campo. Por ejemplo, si se trata de un campo **Hora** podemos escoger la manera en que se guardará (**Figura 9**).

Figura 9. Formato de campo **Hora**.

Para definir cuál (o cuáles) de los atributos de la nueva tabla actuarán como llave primaria para poder especificar de forma inequívoca cada registro de la tabla, nos debemos colocar con el ratón sobre el triángulo verde que señala el campo, hacer clic con el botón derecho y escoger **Llave primaria**. Aparecerá entonces una llave en esta casilla.

Si quisiéramos especificar más de un campo como llave primaria, tendríamos que seleccionar primero los campos deseados haciendo clic sobre la tecla **Control**.

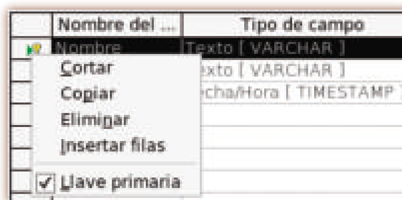


Figura 10.
Especificamos que este campo actuará como **Llave primaria**.

Es habitual utilizar un campo numérico para que actúe como llave primaria de una tabla, y que para cada registro que se añada a la tabla el valor aumente en uno. Para hacerlo, creamos un campo de tipo **Entero (INTEGER)** y especificamos en las propiedades del campo **Valor automático = 'SÍ'** y **Expresión de Autoincremento = 'IDENTITY'**.

Una vez hayamos especificado todos los campos, debemos guardar la tabla haciendo clic en el botón **Guardar** de la *barra de herramientas* **Estándar**.



Figura 11. Botón **Guardar** dentro de la *barra de herramientas* **Estándar**.

Otras operaciones sobre las tablas

Una vez creada la tabla podemos añadirle registros, abriéndola con un doble clic, y escribiendo en cada casilla la información correspondiente.

Figura 12.
Introducción de nuevos registros en una tabla **Autores**.

También es posible editar de nuevo su diseño, cambiarle el nombre o suprimirla definitivamente a través de las opciones del *menú contextual* que aparece si hacemos clic con el botón derecho.

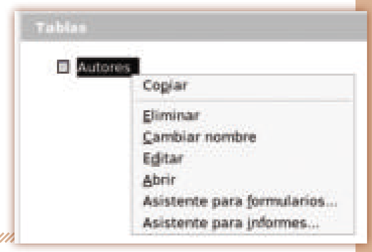


Figura 13.
Operaciones sobre una tabla del menú contextual.

Después de haber creado las tablas es interesante definir cómo se relacionan los diferentes atributos entre éstas. Podemos establecer estas relaciones a través de la opción del menú **Herramientas** | **Relaciones**. Hemos de tener en cuenta que los campos relacionados deben ser obligatoriamente del mismo tipo de datos. En realidad, representan la misma información.

Para establecer una relación simplemente hacemos clic sobre el atributo de una tabla y lo arrastramos hasta el atributo de la otra.

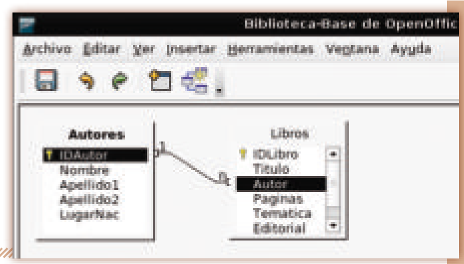


Figura 14.
Relación 1 a muchos entre la tabla **Autores** y la tabla **Libros**: un mismo autor puede haber escrito muchos libros.

Una vez establecida la relación, hacemos doble clic sobre la línea que une los campos para poder acceder a la pantalla donde se especifican ciertas **reglas de integridad**.

Teniendo en cuenta que el campo **idAutor** de la tabla **Autores**, que actúa como llave primaria en esta tabla, se relaciona directamente con el campo **Autor** de la tabla **Libros**, ¿qué pasaría con los libros de *Federico García Lorca* si eliminásemos este autor de la tabla **Autores**? Pues que muchos libros (registros) quedarían huérfanos, ya que no se podría saber quién los escribió.

Solucionar este problema y establecer qué hacer en estas situaciones es el objetivo de esta herramienta (**Figura 14**). El programa nos ofrece cuatro opciones en el caso de que se suprimiera un registro (**Opciones de eliminación**), y algo similar ocurre en el caso de una actualización:



- Si se elimina *Federico García Lorca* de la tabla **Autores**, no se modificará nada de la tabla **Libros** (**Ninguna acción**).

- Se suprimirán todos sus libros (**Eliminar cascada**).

- Pondremos como código de **Autor** un valor null (**Poner null**).

- O, finalmente, pondremos el valor que se haya especificado por defecto en la creación de la tabla (**Predeterminar**).

Figura 15.
Reglas de integridad en una relación.

Generar consultas sobre los datos

Una vez hemos creado las tablas y empezado a incorporar información en ellas, es importante poder realizar consultas sobre éstas. Por ejemplo, si hemos creado una base de datos que contiene información sobre libros, podemos consultar cuáles se publicaron durante el año pasado, cuáles de estos han sido escritos por *Federico García Lorca*, qué libros han obtenido un premio o cuáles de ellos escritos por *Federico García Lorca* obtuvieron un premio.

OOo Base nos ofrece tres posibilidades para hacer la consulta de los datos: **Utilizando el asistente**, en **vista de diseño** y en **vista SQL**, que expondremos a continuación.



Figura 16.
Espacio para trabajar con **Consultas**.

Utilizando el asistente

Para usuarios más noveles, la forma más sencilla de realizar una consulta es utilizando el asistente. A través de este sistema aparece un asistente que nos guía durante ocho pasos.

Paso 1. Selección del campo.
El objetivo de este paso es determinar qué tabla y qué campos de ésta queremos consultar.

Paso 2. Orden de clasificación.
En este paso se establece cuál de los campos determinará el orden de aparición de los registros.

Paso 3. Condiciones de la búsqueda.
En el tercer paso se determinan las condiciones de búsqueda (ver **Figura 17**).

Paso 4. Detalle o resumen.

En este punto es posible seleccionar el tipo de consulta. Si elegimos **Consulta detallada**, nos mostrará los registros que cumplan con las condiciones de búsqueda. En cambio, si escogemos **Consulta abreviada** podemos configurar la consulta para que devuelva el resultado de una función aplicada a los datos. Por ejemplo, buscando el valor máximo o el valor mínimo de un conjunto de registros.

Figura 17.
Asistente de consultas. En este paso especificamos las condiciones de la búsqueda, que mostrará todos los registros de la tabla **Autores** tales que el campo *apellido1* sea igual a García y el campo *apellido2* sea igual a Lorca.

Pasos 5 y 6. Condiciones de agrupación.

En caso que la consulta se haya configurado como **abreviada** aplicando una función en los registros, en estos pasos sería posible determinar una agrupación de los mismos.

Paso 7. Alias.

En este paso podemos otorgar un sobrenombre a los campos de las tablas consultadas. Por ejemplo, establecer el alias **Código Autor** para **idAutor**.

Paso 8. Visión general.

Finalmente, en este último paso determinamos un nombre para la consulta y decidimos qué queremos hacer a continuación (si ver el resultado de la consulta o modificarla).

IDAutor	Nombre	Apellido1	Apellido2	LugarNac
3	Federico	García	Lorca	Fuente Vaqueros

Registro 1 de 1

Figura 18.
Resultado de la consulta generada en la **Figura 17**.

Utilizando la vista de diseño

El primer paso para realizar consultas utilizando este método será escoger con qué tablas existentes en la base de datos queremos efectuar la consulta. Las seleccionamos una a una y hacemos clic en **Añadir** (**Figura 19**).

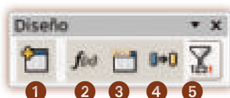


Figura 19. Opciones de la barra de herramientas **Diseño**.

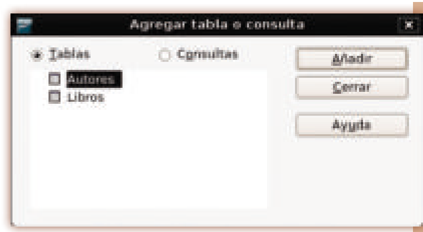


Figura 20. Añadimos tablas para la consulta en la vista de diseño.

A través de **Ver | Barra de herramientas | Diseño** aparecerá la *barra de herramientas* que permite efectuar estas opciones (ver **Figura 19**):

1. Abrir el formulario para añadir las tablas en la consulta en la que estamos trabajando.
2. Activar o desactivar esta casilla muestra u oculta la fila que permite definir funciones.
3. Activar o desactivar esta casilla muestra u oculta la fila que permite escoger la tabla.
4. Activar o desactivar esta casilla muestra u oculta la fila que permite definir un alias para el atributo.
5. Seleccionar esta casilla mostrará en la consulta solamente los registros que contengan valores diferentes (en SQL es la función DISTINCT).

Una vez realizado este paso, en la rejilla inferior se especificarán las condiciones de búsqueda (**Figura 21**). Cada columna de la rejilla especifica una condición para un campo. Los nombres de las filas indican lo siguiente:

- **Campo** ❶: aparece un desplegable para especificar el atributo de la tabla donde queramos aplicar el criterio de búsqueda.
- **Alias** ❷: esta opción nos permite definir un nombre en la cabecera del resultado de la consulta (ejemplo, **Código de autor** en vez de **IdAutor**).
- **Tabla** ❸: tabla donde se efectúa la consulta.
- **Orden** ❹: orden en el que aparecerán los registros que cumplan las condiciones. Podemos poner **Ascendente**, **Descendente** o simplemente no poner nada.
- **Visible** ❺: permite determinar si se mostrará este campo en la consulta resultante.
- **Función** ❻: es posible aplicar funciones en las consultas. Por ejemplo, conocer el número total (*Recuento*) de registros que cumplan cierta condición o, dada una tabla de salarios, el sueldo más alto, el más bajo y el sueldo medio.

Campo ❶	Apellido2
Alias ❷	
Tabla ❸	Autores
Orden ❹	
Visible ❺	☑
Función ❻	
Criterio ❼	='Lorca'
o	
o	
o	

Figura 21. Determinar las condiciones de filtraje de registros.

- **Criterio** ⑦: aquí se aplica el criterio de filtraje de registros por el campo. Por ejemplo, si estamos consultando un campo año, que filtre sólo los que sean del año 2000 (='2000'), los anteriores al 2000 (<'2000') o los posteriores (>'2000').

Utilizando el lenguaje SQL

El Lenguaje Estructurado de Consultas (o *Structured Query Language* (SQL)) es un sistema que permite realizar consultas sobre la base de datos, a parte de insertar, actualizar y eliminar registros.

Aunque no es el objetivo de este fascículo aprender en profundidad este lenguaje, expondremos a continuación algunos ejemplos. Para hacer una consulta de los registros que contiene la tabla **Autores**, escribimos lo siguiente:

SELECT * FROM Autores

Y hacemos clic en el botón **Ejecutar la consulta**.



Figura 22.
Botón **Ejecuta la consulta**, en la *barra de herramientas* **Diseño de la consulta.**

Ahora bien, a través del SQL también podemos aplicar filtros en estas consultas. Esto se hace con la cláusula **WHERE**. Por ejemplo, mostrar los *Autores* que nacieron durante el año 1922:

SELECT * FROM Autors WHERE Año='1922'

O para mostrar los autores que nacieron en el Puerto de Santa María:

SELECT * FROM Autores WHERE lugar_nacimiento='Puerto de Santa María'

O, ambas cosas:

SELECT * FROM Autores WHERE Año='1922' AND lugar_nacimiento='Puerto de Santa María

Los registros resultantes de la consulta se pueden ordenar a través de la cláusula **ORDER BY**. Por ejemplo, todos los autores nacidos en el Puerto de Santa María y ordenando el resultado por su fecha de nacimiento:

SELECT * FROM Autores WHERE
lugar_nacimiento='Puerto de Santa María' ORDER BY fecha_nacimiento

El lenguaje SQL es un estándar que se utiliza en muchos otros sistemas gestores de bases de datos como Microsoft Access, MySQL u ORACLE y hay mucha información de su sintaxis, así como ejemplos, por la Red.

Creación de formularios

Un formulario es una interfaz para entrar y editar los datos de las tablas. En vez de trabajar en un listado de registros, un formulario puede incluir más textos, gráficos, cuadros combinados y muchos otros elementos. Una vez más, existen dos métodos para la creación de formularios.

Uno para no iniciados, más sencillo, que utiliza un **Asistente** y otro, más completo pero más complejo también, utilizando la vista de **Diseño**.



Figura 23.
Espacio para trabajar con **Formularios.**

Si pulsamos sobre **Usar el asistente para crear un formulario**, nos aparecerá un cuadro con los siguientes pasos:

Paso 1.
El primer paso consiste en elegir cuáles de los atributos de las diferentes tablas queremos que aparezcan en el formulario.

Paso 2.
El siguiente paso nos permite añadir un subformulario. Esto serviría para mostrar dentro del mismo formulario tablas que estén relacionadas entre ellas a través de una relación. Si no escogemos añadir ningún otro subformulario en el paso anterior, se salta directamente al paso cinco.

Paso 5.
Sirve para definir cómo se colocarán los controles del nuevo formulario (**Figura 24**).



Figura 24.
Organización de los controles en un formulario utilizando el asistente.

Paso 6.
En el sexto paso podemos especificar para qué se utilizará el nuevo formulario. Si servirá sólo para introducir nuevos registros de información o si servirá también para mostrar datos.

Paso 7.

En el séptimo paso podemos escoger alguno de los estilos que están predefinidos en el programa, con un color de fondo para el formulario y el estilo del contorno de los controles.

Ya sólo nos queda, como último paso, definir un nombre para el nuevo formulario. Una vez hemos creado el formulario, podemos trabajar con él haciendo clic sobre el nombre dentro del apartado **Formularios**.

> En OpenOffice.org tanto los formularios para el acceso a las bases de datos como los informes se generan utilizando el módulo Writer.

Utilizando la vista de diseño

Para trabajar con este método hacemos clic en **Crear un formulario en vista de diseño....** Observaremos que se abrirá OpenOffice.org Writer con la *barra de herramientas* **Campos de control de formulario** activada y en **Modo de diseño** (ver **Figura 25**).



Figura 25.
Barra de herramientas **Campos de control de formulario** y marcado en rojo el botón destinado a activar/desactivar el **Modo diseño**.

Una vez hayamos insertado un primer control, es importante definir las propiedades del formulario para especificar de qué tabla o consulta extraeremos los datos (ver **Figura 26**).

Para esto, hacemos clic en el botón **Formulario** dentro de la *barra de herramientas* **Campos de control de formulario**.

Ahora, ya es posible añadir los controles que queramos en el formulario. Tenemos la posibilidad de añadir cuadros de texto, cuadros de verificación, cuadros de lista, etc... Para cada control podemos especificar qué atributo de las tablas representa. Para ello, hacemos clic con el botón derecho sobre el control correspondiente, pulsamos **Control**, pestaña **Datos**, campo **Campo de datos**.

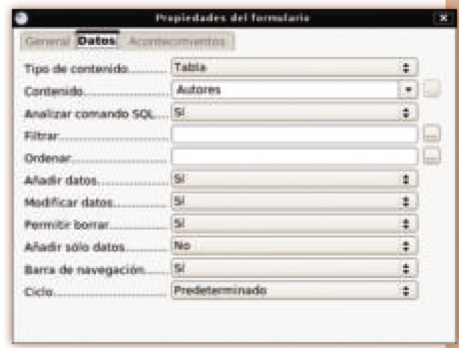


Figura 26. Propiedades del formulario.

> Si queremos cambiar el color de fondo de un formulario debemos hacer clic en el botón derecho para que aparezca el *menú contextual*, escogemos la opción **Página** y hacemos clic en la pestaña **Fondo**.

El botón **Orden de activación** de la *barra de herramientas* **Diseño de formularios** sirve para especificar el orden de tabulación (la manera en la que se salta de control a control utilizando el tabulador).

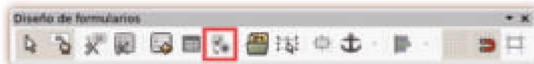


Figura 27.
Botón **Orden de Activación** de la *barra de herramientas* **Diseño de formularios**.

Generación de informes

Es posible extraer informes de los datos de nuestra base de datos utilizando este apartado. Los informes se generan a partir de tablas o consultas de la base de datos y pueden contener todos los campos de una tabla o sólo el que deseemos.

Los informes pueden ser estáticos o dinámicos. Los primeros contendrán siempre los datos presentes en la tabla o consulta en el momento de la creación del informe, mientras que los informes dinámicos mostrarán siempre los datos que haya en el momento de la visualización del informe.



Figura 28.
Espacio para trabajar con **Informes**.

Para generar un informe hacemos clic en **Crear un informe utilizando el asistente...**. A continuación nos aparecerá un asistente con los pasos siguientes:

Paso 1. Selección del campo.
Antes de todo debemos seleccionar de qué campos de las tablas queremos generar el informe.

Paso 2. Se están etiquetando los campos.
En el siguiente paso es posible definir una etiqueta para el campo de la tabla.

Paso 3. Agrupación.
En este paso podemos establecer niveles de agrupación en nuestros datos.

Paso 4. Opciones de ordenación.
Elegimos en qué orden queremos que aparezcan los registros en nuestro informe (ver **Figura 29**).

Paso 5. Escoger un formato.
Escogemos el formato y el estilo del informe. También podemos definir la orientación del informe resultante (apaisado o vertical). No obstante, hemos de tener en cuenta que estas opciones se pueden cambiar más adelante.

Paso 6. Crear un informe.
Estamos ya en el último paso. En este cuadro especificaremos el título del informe, en el **tipo de informe que queremos crear** es posible escoger si los datos del informe serán estáticos (una vez generado, estos datos ya no cambiarán) o dinámicos (creándolo siempre a partir de los datos de la base de datos) y la actuación a realizar después de crear el informe.



Figura 29.
Opciones de ordenación en el asistente de creación de informes.

Otros aspectos avanzados de OOo Base

Desde otras aplicaciones de OpenOffice.org como Writer o Calc es posible acceder a las fuentes de datos registradas creadas usando el módulo Base a través de **Ver** | **Fuentes de datos F4**.

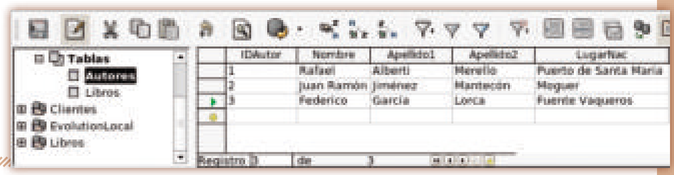
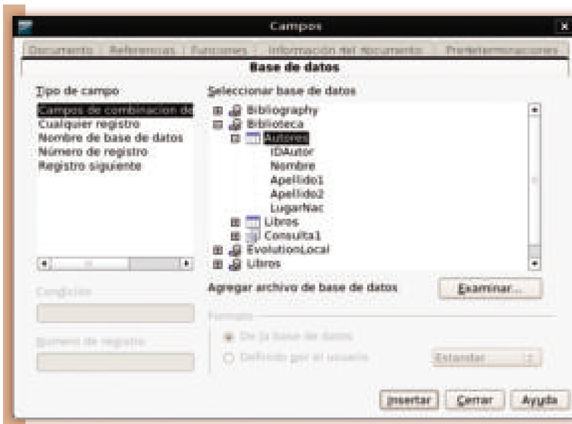


Figura 30.
Acceso a las fuentes de datos a través de OpenOffice.org Calc.



Desde OOo Writer podemos introducir datos procedentes de estas bases de datos en nuestros documentos a través de la opción del menú **Insertar** | **Campos** | **Otros** | **Base de datos**.

Figura 31.
Inserción de campos desde un documento en Writer.

Una aplicación típica del uso conjunto del editor de textos Writer y Base sería poder crear un carta (o mensaje de correo electrónico) personalizada para cada uno de los destinarios presentes en una tabla de una base de datos.

Imaginemos que tenemos una tabla con información y direcciones de diversos amigos, conocidos, clientes o socios con algunos campos similares a los siguientes:

Clientes-Base

Archivo Editar Ver Herramientas Ventana Ayuda

Nombre del ...	Tipo de campo
IDAmigo	Integer [INTEGER]
nombre	Texto [VARCHAR]
apellido1	Texto [VARCHAR]
apellido2	Texto [VARCHAR]
direccion	Texto [VARCHAR]
localidad	Texto [VARCHAR]
cotpostal	Texto [VARCHAR]
provincia	Texto [VARCHAR]
telefono	Texto [VARCHAR]

Figura 32.
Estructura de una tabla con información y datos de contacto.

Ahora, desde Writer podríamos escribir una carta personalizada para cada uno de ellos a través de la opción **Herramientas | Asistente para combinar correspondencia**. El programa iniciará un asistente para guiar al usuario durante todo el proceso.

Paso 1. Selección del documento de inicio.

El primer paso consiste sólo en seleccionar el documento que utilizaremos para redactar la carta maestra. Podemos partir del documento actual o crear uno nuevo, entre otras opciones.



Figura 33.
Primer paso en el asistente para combinar correspondencia entre Writer y Base.

Paso 2. Selección del tipo de documento.

El siguiente paso es indicar si queremos crear una combinación para generar cartas personalizadas o mensajes de correo electrónico. En este caso utilizamos la opción **Carta**.



Figura 34.
Segundo paso en el asistente: decidimos qué tipo de documento queremos crear.

Paso 3. Insertar una agenda de direcciones.
El tercer paso sirve para indicar qué campos queremos que se muestren en la carta generada. Recordemos que, como estamos extrayendo estos datos de una base de datos, tenemos que hacer coincidir los campos de la agenda con los definidos en la tabla de la base de datos.



Figura 35.
Herramienta para establecer las coincidencias entre los campos de la agenda de direcciones y los campos de la base de datos.

Paso 4. Crear el saludo.
A través del cuarto paso creamos un saludo personalizado (que puede ser diferente, por ejemplo, si se trata de un destinatario masculino o femenino).

Paso 5. Ajustar el formato.
El siguiente paso en el asistente nos indica cómo ajustar el formato del documento resultante.



Figura 36.
El quinto paso sirve para realizar ajustes en el formato resultante.

Paso 6. Editar el documento.
Éste es el momento en el que realizaremos definitivamente la combinación entre el documento maestro y los diversos registros de la base de datos, haciendo la sustitución en cada campo del documento maestro para cada valor del campo de la tabla.

Paso 7. Personalizar el documento.
Aquí se nos ofrece la posibilidad de reeditar y personalizar el documento una vez se ha hecho la combinación.

Paso 8. Guardar, imprimir o enviar.
Se trata del último paso y es en el que hemos de decidir qué queremos hacer con el documento generado.

Si lo que nos interesa ahora es enviar los documentos combinados por correo, hemos de definir antes los datos del servidor de correo saliente (SMTP).

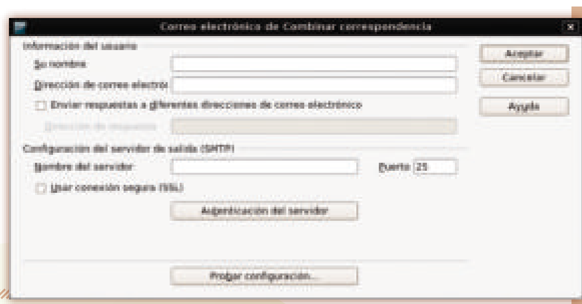


Figura 37.
Configuración del servidor de correo electrónico saliente.

cenatic

CENATIC es el Centro Nacional de Referencia de Aplicación de las TIC basadas en Fuentes Abiertas, una Fundación Pública Estatal dependiente del Ministerio de Industria, Turismo y Comercio encargada de la difusión y la promoción del Software de Fuentes abiertas en todos los ámbitos de la sociedad. Somos una Fundación Pública Estatal promovida por el Ministerio de Industria, Turismo y Comercio (a través de la Secretaría de Estado de Telecomunicaciones y para la Sociedad de la Información y la entidad pública *red.es*) y la Junta de Extremadura, que además cuenta en su Patronato con los gobiernos de Andalucía, Asturias, Aragón, Cantabria, Cataluña e Islas Baleares. También forman parte del Patronato de CENATIC las empresas Atos Origin, Sun Microsystems, Bull, Telefónica y Gpex. Puedes saber más sobre nosotros en *http://www.cenatic.es*

Autor: **Gerard Farràs i Ballabriga**
Traducción de la obra original: **Francesc Dorca i Badia**
Adaptación de la obra derivada: **CENATIC**
Responsable del proyecto: **CENATIC**
Basado en una adaptación de la obra de:



Consejería de Innovación, Ciencia y Empresa de la Junta de Andalucía

Obra original de:



Regidoria Ciutat del Coneixement de l'Ajuntament de Barcelona

Servicios editoriales y traducción: **Edit Lin Editorial, S.L.**

